This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

[CLAIMS] What is claimed:

1.   (Original) A method of compressing a data file comprising a sequence of bytes of a length greater than or equal to a predefined length, the method including the steps of:
    retrieving the data file from a secondary storage device;
    storing the data file in direct access memory;
    calculating the frequency of unique byte values within a sub-sequence of the data file, the sub-sequence having a length not exceeding the predefined length;
    creating an index for the sub-sequence, the index including a data value representing the calculated frequency of unique byte values within the sub-sequence; and
    on the sub-sequence having a frequency of unique byte values below a predetermined threshold, applying a data transformation to the sub-sequence to increase the frequency of unique byte values in the sub-sequence and adding to the index a data value representing the data transformation;
    on the sub-sequence having a frequency of unique byte values above a predefined threshold, adding to the index a data value representing the position of one or more unique values within the sub-sequence;
    creating an output data file, the data file having a file type identifier; and
    adding the index to the output data file.

2.   (Original) A method of compressing a data file as claimed in claim 1 wherein the step of applying a data transformation to the sub-sequence further includes the steps of:
    maintaining in computer memory a plurality of transformation data sets, the data sets having a sequence of byte values and identified by a transformation data set identifier;
    retrieving one of the transformation data sets from computer memory, the retrieved transformation data set having a length substantially equal to the length of the sub-sequence of the data file; and
    applying a data transformation to respective byte values in the sub-sequence based on corresponding byte values in the retrieved data set.

3.   (Original) A method of compressing a data file as claimed in claim 2 wherein the sub-sequence following data transformation based on at least one of the retrieved transformation data set is substantially identical to the sub-sequence prior to the data transformation.

4.   (Original) A method of compressing a data file as claimed in claim 2 wherein at least one of the transformation data sets comprises a randomly generated sequence of byte rates.

5.   (Original) A method of compressing a data file as claimed in claim 2 wherein at least one of the transformation data sets comprises a predefined sequence of byte rates.

6.   (Original) A method of compressing a data file as claimed in claim 2 wherein at least one of the transformation data sets comprises a sequence of byte values derived from a portion of the data file other than the sub-sequence of the data file.

7.   (Currently amended) A method of compressing a data file as claimed in [any one of claims 2 to 6] claim 2 further comprising the step of adding to the index the transformation data set identifier of the data transformation data set applied to the sub-sequence.

8.   (Currently amended) A method of compressing a data file [as claimed in any one of the preceding claims further comprising the step of calculating the position of the one or more unique values within the sub-sequence.] comprising a sequence of bytes of a length greater than or equal to a predefined length, the method including the steps of:
        retrieving the data file from a secondary storage device;
        storing the data file in direct access memory;
        calculating the frequency of unique byte values within a sub-sequence of the data file, the sub-sequence having a length not exceeding the predefined length;
        calculating the position of the one or more unique values within the sub-sequence;
        creating an index for the sub-sequence, the index including a data value representing the calculated frequency of unique byte values within the sub-sequence; and
        on the sub-sequence having a frequency of unique byte values below a predetermined threshold, applying a data transformation to the sub-sequence to increase the frequency of unique byte values in the sub-sequence and adding to the index a data value representing the data transformation;
        on the sub-sequence having a frequency of unique byte values above a predefined threshold, adding to the index a data value representing the position of one or more unique values within the sub-sequence;
        creating an output data file, the data file having a file type identifier; and
        adding the index to the output data file.

9.   (Original) A method of compressing a data file as claimed in claim 8 wherein the step of calculating the position of the one or more unique values within the sub-sequence further comprises the steps of:

creating a temporary position index in computer memory;

retrieving successive byte values from the sub-sequence;

on retrieving each byte value, determining whether the retrieved byte value is a unique byte value or a repeated value;

on detection of a unique byte value, adding one of two bit values to the temporary position index otherwise adding the other of the two bit values to the temporary position index;

creating a position index representing the position of the one or more unique values from the temporary position index; and

calculating the data value representing the position of the one or more unique values at least partly from the position index.

10.   (Original) A method of compressing a data file as claimed in claim 9 wherein the number of bytes in the sub-sequence is substantially equal to the number of bits in the temporary position index.

11.   (Currently amended) A method of compressing a data file as claimed in claim 9 [or claim 10] wherein the size of the position index is less than the size of the temporary position index.

12.   (Currently amended) A method of compressing a data file as claimed in [any one of claims 9-11] claim 9 further comprising the steps of:

creating a permutation index representing the order of unique byte values within the sub-sequence; and

calculating the data value representing the position of the one or more unique values both from the position index and the permutation index.

13.   (Original) A method of compressing a data file as claimed in claim 12 comprising the step of concatenating the position index and the permutation index to form the data value representing the position of the one or more unique values.

14.   (New) A system for compressing a data file comprising a sequence of bytes of a length greater than or equal to a predefined length, where the system is configured to:

retrieve the data file from a secondary storage device;

store the data file in direct access memory;

calculate the frequency of unique byte values within a sub-sequence of the data file, the sub-sequence having a length not exceeding the predefined length;

create an index for the sub-sequence, the index including a data value representing the calculated frequency of unique byte values within the sub-sequence; and

on the sub-sequence having a frequency of unique byte values below a predetermined threshold, apply a data transformation to the sub-sequence to increase the frequency of unique byte values in the sub-sequence and add to the index a data value representing the data transformation;

on the sub-sequence having a frequency of unique byte values above a predefined threshold, add to the index a data value representing the position of one or more unique values within the sub-sequence;

create an output data file, the data file having a file type identifier; and

add the index to the output data file.

15.   (New)   A system for compressing a data file comprising a sequence of bytes of a length greater than or equal to a predefined length, where the system is configured to:

retrieve the data file from a secondary storage device;

store the data file in direct access memory;

calculate the frequency of unique byte values within a sub-sequence of the data file, the sub-sequence having a length not exceeding the predefined length; and

calculate the position of the one or more unique values within the sub-sequence;

create an index for the sub-sequence, the index including a data value representing the calculated frequency of unique byte values within the sub-sequence; and

on the sub-sequence having a frequency of unique byte values below a predetermined threshold, apply a data transformation to the sub-sequence to increase the frequency of unique byte values in the sub-sequence and add to the index a data value representing the data transformation;

on the sub-sequence having a frequency of unique byte values above a predefined threshold, add to the index a data value representing the position of one or more unique values within the sub-sequence;

create an output data file, the data file having a file type identifier; and
add the index to the output data file.

16.   (New)   A computer program stored on tangible storage medium comprising executable instructions for performing a method of compressing a data file comprising a sequence of bytes of a length greater than or equal to a predefined length, the method comprising:

retrieving the data file from a secondary storage device;

storing the data file in direct access memory;

calculating the frequency of unique byte values within a sub-sequence of the data file, the sub-sequence having a length not exceeding the predefined length;

creating an index for the sub-sequence, the index including a data value representing the calculated frequency of unique byte values within the sub-sequence; and

on the sub-sequence having a frequency of unique byte values below a predetermined threshold, applying a data transformation to the sub-sequence to increase the frequency of unique byte values in the sub-sequence and adding to the index a data value representing the data transformation;

on the sub-sequence having a frequency of unique byte values above a predefined threshold, adding to the index a data value representing the position of one or more unique values within the sub-sequence;

creating an output data file, the data file having a file type identifier; and

adding the index to the output data file.

17.   (New)   A computer program stored on tangible storage medium comprising executable instructions for performing a method of compressing a data file comprising a sequence of bytes of a length greater than or equal to a predefined length, the method comprising:

retrieving the data file from a secondary storage device;

storing the data file in direct access memory;

calculating the frequency of unique byte values within a sub-sequence of the data file, the sub-sequence having a length not exceeding the predefined length; and

calculating the position of the one or more unique values within the sub-sequence;

creating an index for the sub-sequence, the index including a data value representing the calculated frequency of unique byte values within the sub-sequence; and

on the sub-sequence having a frequency of unique byte values below a predetermined threshold, applying a data transformation to the sub-sequence to increase the frequency of unique byte values in the sub-sequence and adding to the index a data value representing the data transformation;

on the sub-sequence having a frequency of unique byte values above a predefined threshold, adding to the index a data value representing the position of one or more unique values within the sub-sequence;

creating an output data file, the data file having a file type identifier; and

adding the index to the output data file.